LINUX PROGRAMMING LABORATORY

VI Semester: CSE / IT

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
AITB13	**Core**	-	-	2	1	30	70	100
Contact Classes: Nil	**Tutorial Classes: Nil**	**Practical Classes: 24**				**Total Classes: 24**		

I. COURSE OVERVIEW:

This course covers operating system concepts in linux environment. It focuses on practice on shell commands and demonstration of process concepts such as creation and establishing communication using linux system calls. The main objective of the course is to teach the students how to work with linux environment and demonstration of operating systems concepts using linux system calls in C programs. This course reaches to student by power point presentations, lecture notes, and lab which involve the problem solving in mathematical and engineering areas.

II. OBJECTIVES:

The course should enable the students to:

I. Familiar with the Linux command-line environment.

II. Understand system administration processes by providing a hands-on experience.

III. Understand Process management and inter-process communications techniques.

III. COURSE OUTCOMES:

After successful completion of the course, students should be able to:

CO 1	**Demonstrate** text processing utilities, file handling utilities, security by file permissions, process utilities, disk utilities and networking commands with different options available for solving problems.	Understand
CO 2	**Make use of** bourne shell constructs, decision structures and loops in designing programs for complex problems.	Apply
CO 3	**Interpret** to write, compile, debug and run C language program in linux shell environment for implementing kernel level concepts.	Understand
CO 4	**Identify** basic methods and techniques used in solving simple programming tasks in the area of execution environment, processessignals and threads.	Apply
CO5	**Experiment with** IPC mechanisms such as pipes, named pipes, shared memory, message queues, semaphores and sockets for interprocess communication.	Apply
CO 6	**Choose** the appropriate protocol such as TCP or UDP for effective communication in client-server applications.	Apply

IV. SYLLABUS:

LIST OF EXPERIMENTS

Week-1	**BASIC COMMANDS I**
Study and Practice on various commands like man, passwd, tty, script, clear, date, cal, cp, mv, ln, rm, unlink, mkdir, rmdir, du, df, mount, umount, find, unmask, ulimit, ps, who, w.	
Week-2	**BASIC COMMANDS II**
Study and Practice on various commands like cat, tail, head , sort, nl, uniq, grep, egrep,fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, tar, cpio.	
Week-3	**SHELL PROGRAMMING, I**
a) Write a Shell Program to print all .txt files and .c files. b) Write a Shell program to move a set of files to a specified directory. c) Write a Shell program to display all the users who are currently logged in after a specified time. d) Write a Shell Program to wish the user based on the login time.	

<table>
<tr><td><b>Week-4</b></td><td><b>SHELL PROGRAMMING II</b></td></tr>
<tr><td colspan="2">a) Write a Shell program to pass a message to a group of members, individual member and all.
b) Write a Shell program to count the number of words in a file.
c) Write a Shell program to calculate the factorial of a given number.
d) Write a Shell program to generate Fibonacci series.</td></tr>
<tr><td><b>Week-5</b></td><td><b>SIMULATING COMMANDS I</b></td></tr>
<tr><td colspan="2">a) Simulate cat command b) Simulate cp command</td></tr>
<tr><td><b>Week-6</b></td><td><b>SIMULATING COMMANDS II</b></td></tr>
<tr><td colspan="2">a) Simulate tail command b) Simulate head command</td></tr>
<tr><td><b>Week-7</b></td><td><b>SIMULATING COMMANDS III</b></td></tr>
<tr><td colspan="2">a) Simulate mv command b) Simulate nl command</td></tr>
<tr><td><b>Week-8</b></td><td><b>SIGNAL HANDLING</b></td></tr>
<tr><td colspan="2">Write a program to handle the signals like SIGINT, SIGDFL, SIGIGN</td></tr>
<tr><td><b>Week-9</b></td><td><b>INTERPROCESS COMMUNICATIONS</b></td></tr>
<tr><td colspan="2">Implement the following IPC forms
a) FIFO b) PIPE</td></tr>
<tr><td><b>Week-10</b></td><td><b>MESSAGE QUEUES</b></td></tr>
<tr><td colspan="2">1. Write a C program (sender.c) to create a message queue with read and write permissions to write 3 messages to it with different priority numbers.
2. Write a C program (receiver.c) that receives the messages (from the above message queue as specified and displays them.</td></tr>
<tr><td><b>Week-11</b></td><td><b>SHARED MEMORY</b></td></tr>
<tr><td colspan="2">Implement shared memory form of IPC.</td></tr>
<tr><td><b>Week-12</b></td><td><b>SOCKET PROGRAMMING</b></td></tr>
<tr><td colspan="2">1. Write client and server programs (using c) for interaction between server and client processes using TCP Elementary functions.
2. Write client and server programs (using c) for interaction between server and client processes using UDP Elementary functions.</td></tr>
<tr><td colspan="2"><b>Reference Books:</b></td></tr>
<tr><td colspan="2">1. Sumitabha Das, “Your Unix The Ultimate Guide”, Tata McGraw-Hill, New Delhi, India, 2007.
2. B. A. Forouzan and R. F. Gilberg, “Unix and Shell Programming”, Cengage Learning.
3. Robert Love, “Linux System Programming”, O'Reilly, SPD.
4. Stephen G. Kochan, Patrick Wood, “Unix Shell Programming”, Sams publications, 3rd Edition, 2007.
5. T. Chan, “Unix System Programming using C++”, Prentice Hall India, 1999.</td></tr>
<tr><td colspan="2"><b>Web References:</b></td></tr>
<tr><td colspan="2">1. http://spoken-tutorial.org/tutorial search/?search_foss=Linux&search_language=English
2. https://www.redhat.com/en/files/resources/en-rhel-whats-new-in-rhel-712030417.pdf
3. http:// www.tutorialspoint.com/unix/
4. http://cse09-iiith.virtual-labs.ac.in/</td></tr>
<tr><td colspan="2"><b>SOFTWARE AND HARDWARE REQUIREMENTS FOR A BATCH OF 36 STUDENTS:</b>
<b>HARDWARE:</b> Desktop Computer Systems: 36nos
<b>SOFTWARE:</b> System Software: Linux Operating System</td></tr>
</table>